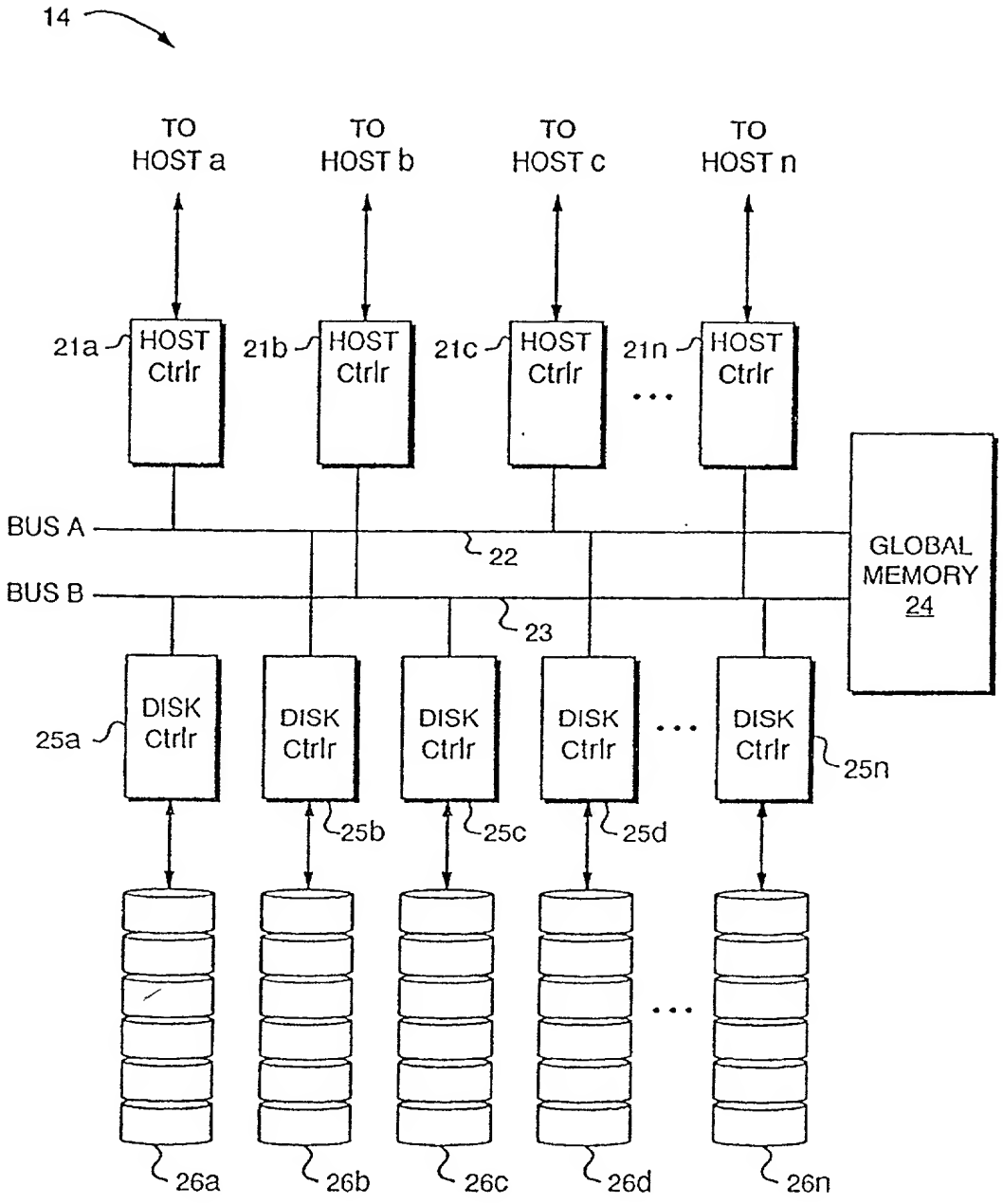
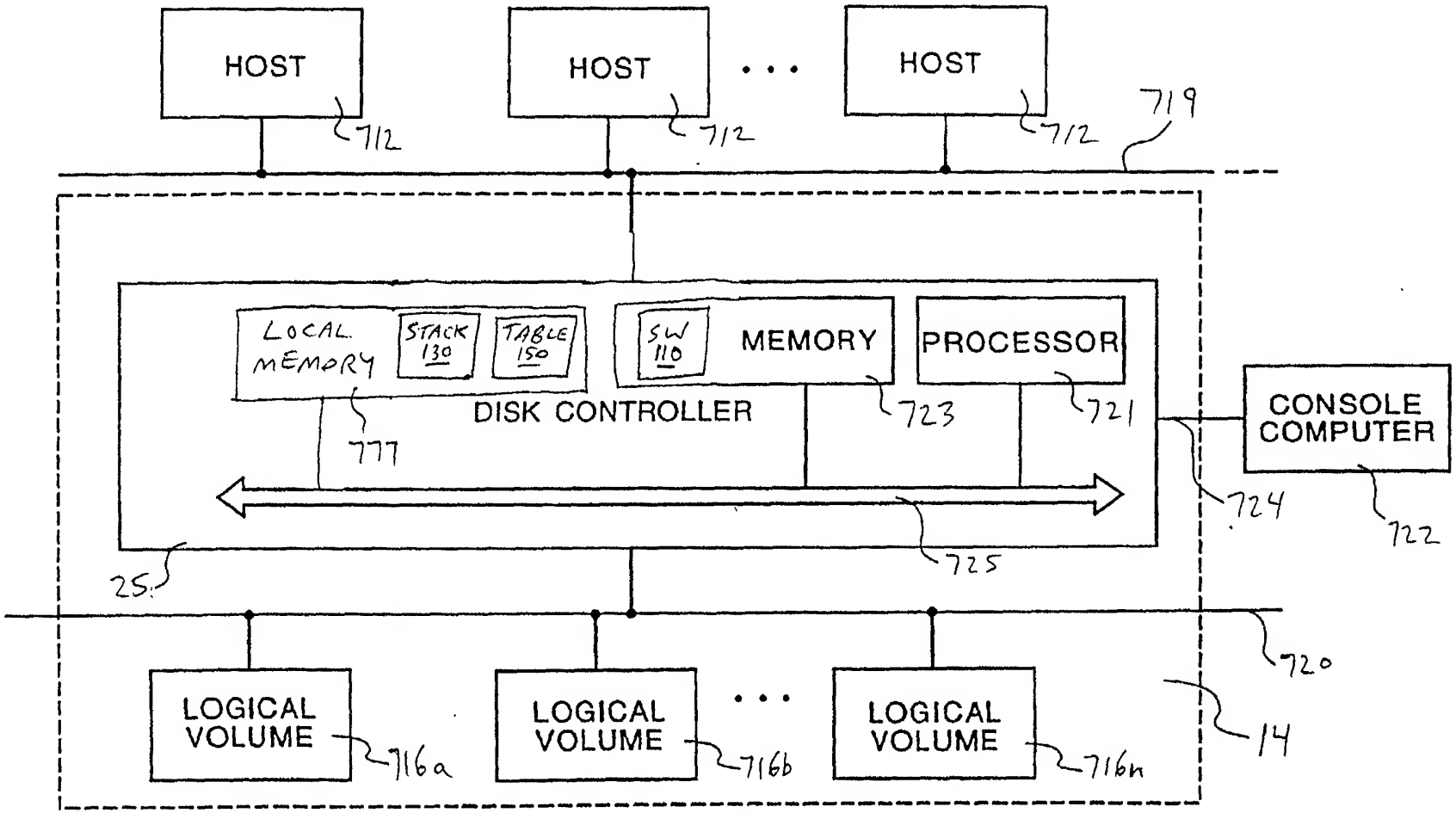

14
TO HOST a
TO HOST b
TO HOST c
TO HOST n
21a
HOST Ctrlr
21b
HOST Ctrlr
21c
HOST Ctrlr
21n
HOST Ctrlr
BUS A
22
BUS B
23
GLOBAL MEMORY 24
25a
DISK Ctrlr
DISK Ctrlr
DISK Ctrlr
DISK Ctrlr
DISK Ctrlr
25n
25b
25c
25d
26a
26b
26c
26d
26n

FIG. 1

FIG. 2



```
LOCATE_AT(PROFILE_DATA_area)
EXTERN T_SYMPROF_TABLE_ENTRY symprof_stat_table [SYMPROF_MAX_TABLE_ENTRIES];
END_LOCATION



LOCATE_AT(LCLMM_SYM_PROFILER)
EXTERN T_PROFILER_SETUP_RECORD     profilerSetupTable [PROFILER_RANGES_Q];
EXTERN TCO_PROFILER_RUNTIME_INFO   runtimeInfo;
EXTERN ULONG                       symprof_stack_index;    /* index into the profiler stack frame */
EXTERN TCO_SYMPROF_STACK_FRAME     symprof_stack_base [MAX_NUMBER_OF_SYM_PROF_STACK_FRAMES];
END_LOCATION



typedef enum
{
   PROFILER_MODE_UNDEFINED = 0,
   PROFILER_MODE_PROFILING = 1,
   PROFILER_MODE_TRACING   = 2,
   PROFILER_MODE_ONCE      = 3
} T_PROFILER_MODE;
```

FIG. 3

T_SYMPROF_TABLE_ENTRY 310

MSB LSB

Self Time (64 bits)	314
Total Time (64 bits)	316
Number of calls (64 bits)	318
Function Address (32 bits) 312	**Profiler instruction address** (32 bits) 320

FIG. 4

410

T_SYMPROF_STACK_FRAME

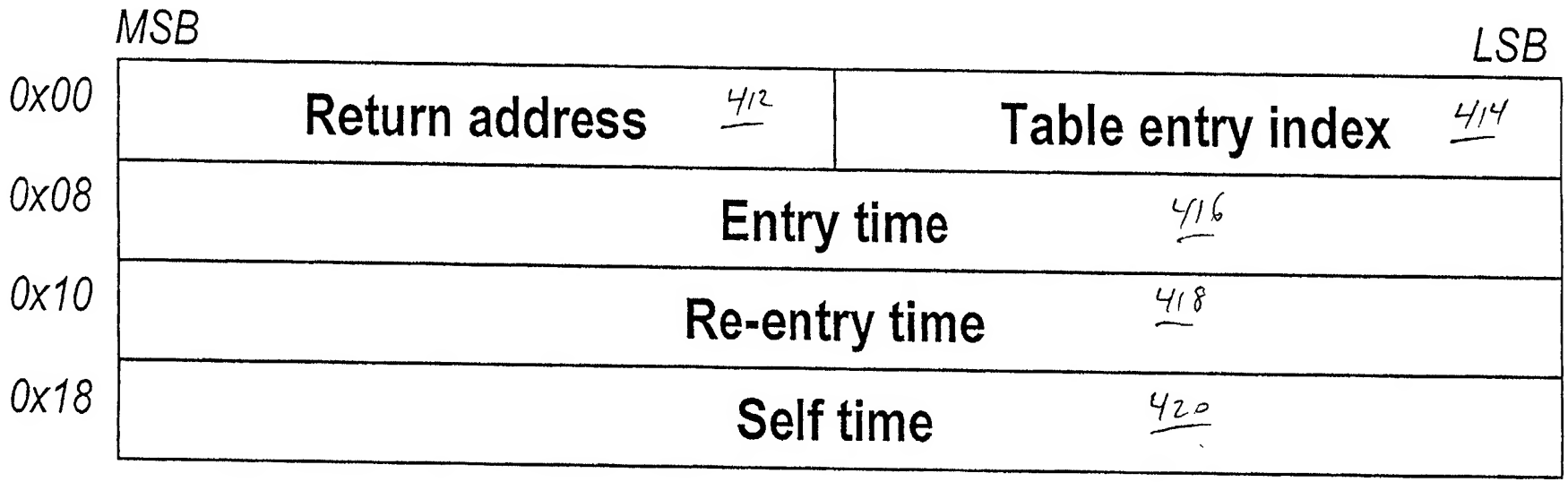


FIG. 5

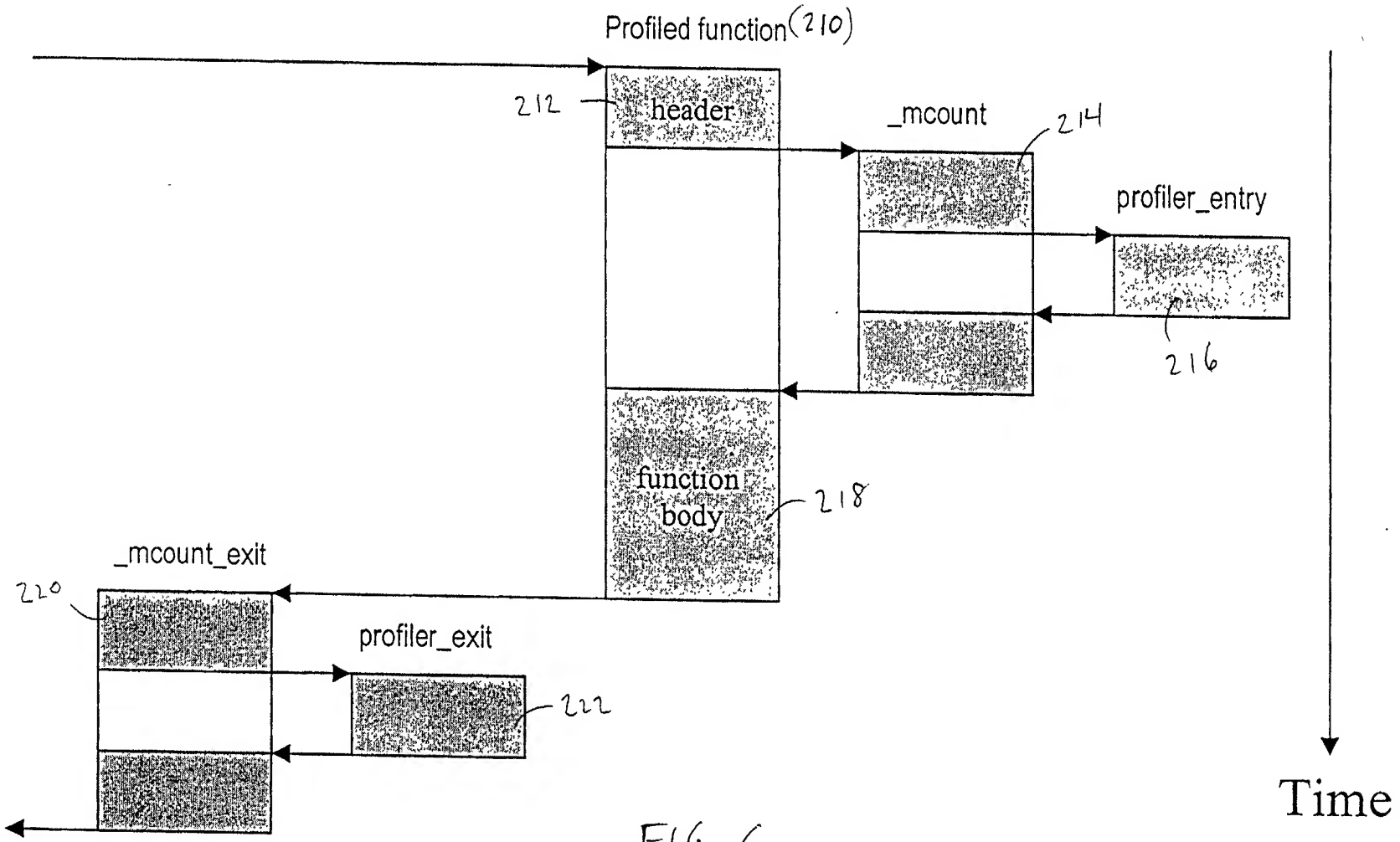

Profiled function (210)
212
header
_mcount
214
profiler_entry
216
function body
218
_mcount_exit
220
profiler_exit
222
Time

FIG. 6

Original program code

```
BOOLEAN update_restored_ckd_id_table(T_DV_REC *p_dv)
{
   stwu %r1,-16(%r1)
   mflr %r0
   stw %r31,12(%r1)
   stw %r0,20(%r1)
   mr %r31,%r3
```

With Profiler code:

```
BOOLEAN update_restored_ckd_id_table(T_DV_REC *p_dv)
{
   stwu %r1,-16(%r1)
   mflr %r0
   stw %r31,12(%r1)
   stw %r0,20(%r1)
   .section ".bss1"
   .align 2
   .long 0
   .long 0
   .long 0
   .long 0
   .long 0
   .long 0
   .long 0
   .long 0
   .long 0
   .section ".text"
   b .+8    <<--   Replaced by "bla _mcount" when profiling enabled
   .long .LP0
   mr %r31,%r3
```

510

FIG. 7

```
void  profiler_entry (T_SYMPROF_TABLE_ENTRY* self_entry, ULONG* params_p)
{
   UINT64 time_now = getTimeBase(); /* Take a snapshot of the current time */

   /* Stop the profiler if we are over the time limit */
   if (runtimeInfo.timeLimit &&
       ((cu_timer - runtimeInfo.startTime) > runtimeInfo.timeLimit))
   {
      /* Over the time limit - stop the profiler */
      BIT_CLEAR(B_SYMPROF_ON, runtimeInfo.flags);
      runtimeInfo.stopTime = cu_timer;
   }

   /* Start the timer for the function being profiled (self) and reset the
      self total time */

   symprof_stack_base [symprof_stack_index].entry_time   = time_now;
   symprof_stack_base [symprof_stack_index].reentry_time = time_now;
   symprof_stack_base [symprof_stack_index].self_time    = 0;

   /* Set the index to which this stack frame refers */
   symprof_stack_base [symprof_stack_index].table_entry = (ULONG) self_entry;

   /* Stop the timer of the parent function (parent) */
   if (symprof_stack_index != 0)
   {
      /* We are not at top level of the profiler stack - compute the total
         time spent in the parent since last reentry to the parent */

      symprof_stack_base[symprof_stack_index-1].self_time +=
         time_now - symprof_stack_base[symprof_stack_index-1].reentry_time;
   }

   /* Increment self counter */
   self_entry->calls++;

   /* Push self stack frame on profiler stack, check for overflow */
   if (++symprof_stack_index >= MAX_NUMBER_OF_SYM_PROF_STACK_FRAMES)
   {  /* Profiler stack overflow: disable profiling */
      BIT_CLEAR(B_SYMPROF_ON, runtimeInfo.flags);
      runtimeInfo.stopTime = cu_timer;
   }

   if (runtimeInfo.mode == PROFILER_MODE_TRACING)
   {
      traceRoutineEntry (self_entry->functionAddr, params_p);
   }
}
```

FIG. 8

```
void  profiler_exit (ULONG* params_p)
{
   T_SYMPROF_TABLE_ENTRY* self_entry;
   UINT64  time_now = getTimeBase (); /* Take a snapshot of the current time */

   /* Pop the calling function's stack frame from profiler's stack */
   symprof_stack_index--;

   /* Update the re-entry time of the parent function */
   if (symprof_stack_index != 0)
   {
      /* We are not at top level of the profiler stack - store the reentry
         time of the parent function */
      symprof_stack_base [symprof_stack_index-1].reentry_time = time_now;
   }

   /* Don't update the entry if the profiler is off */
   if (BIT_IS_CLEAR(B_SYMPROF_ON, runtimeInfo.flags))
   {
      return;
   }

   /* Update the current function's profiler table entry */
   self_entry = (T_SYMPROF_TABLE_ENTRY *)
      symprof_stack_base[symprof_stack_index].table_entry;

   self_entry->self_time +=
      TimeBase2Micros(time_now -
symprof_stack_base[symprof_stack_index].reentry_time +
                      symprof_stack_base[symprof_stack_index].self_time);

   self_entry->total_time +=
      TimeBase2Micros(time_now -
symprof_stack_base[symprof_stack_index].entry_time);

   if (runtimeInfo.mode == PROFILER_MODE_TRACING)
   {
      traceRoutineExit (self_entry->functionAddr, params_p);
   }

   if (runtimeInfo.mode == PROFILER_MODE_ONCE)
   { /* Disable profiling by replacing "bla _mcount" with "br .+8" */
     insertInstruction ((ULONG *)self_entry->profilerInstructionAddr,
PPC_B_LOCATION_PLUS_8_INSTR);
   }
}
```

FIG. 9

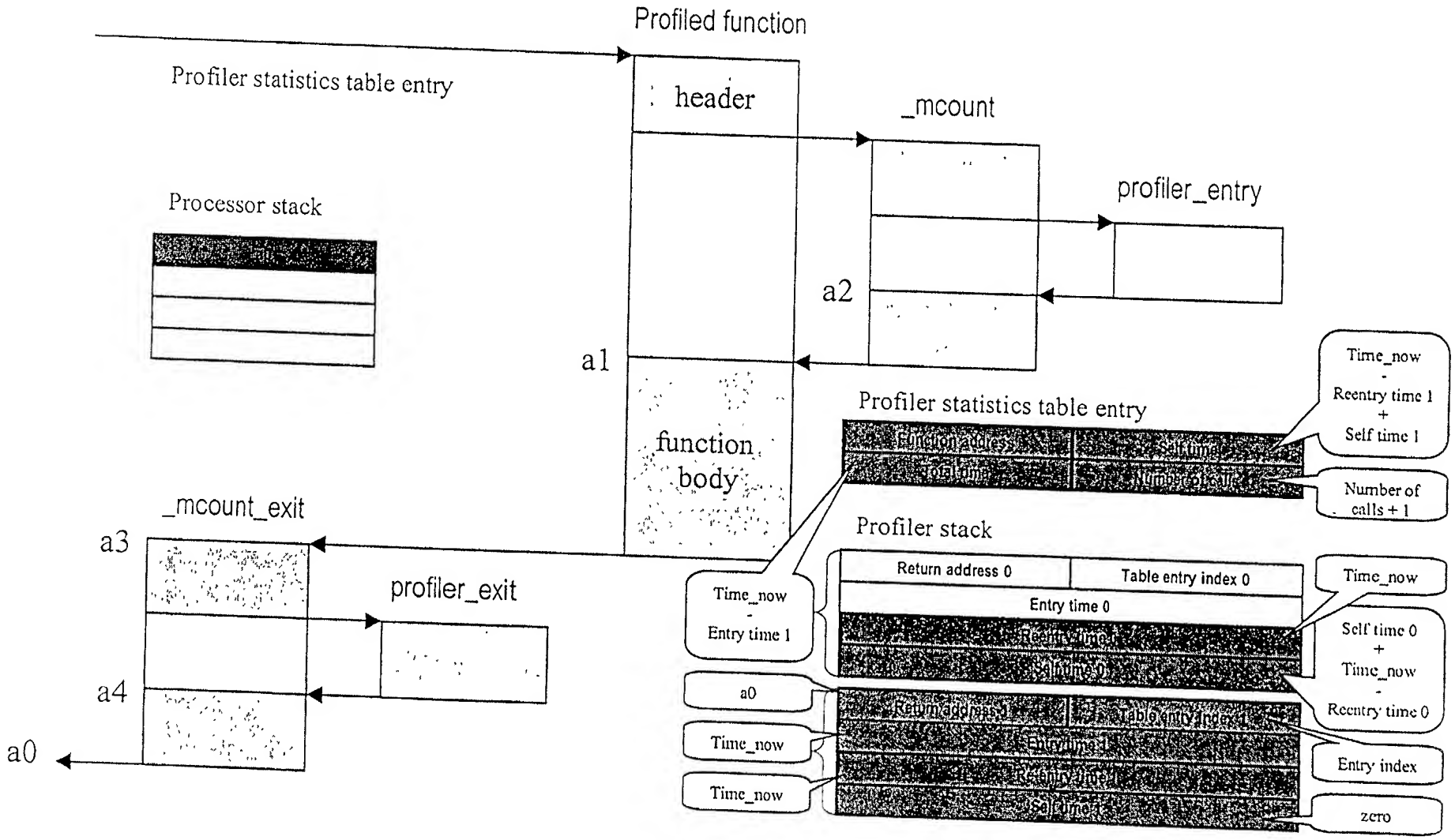

Profiled function
Profiler statistics table entry
header
_mcount
profiler_entry
Processor stack
a2
a1
function body
Time_now - Reentry time 1 + Self time 1
Profiler statistics table entry
Function address
Self time
Total time
Number of calls
Number of calls + 1
_mcount_exit
a3
Profiler stack
Return address 0
Table entry index 0
Time_now - Entry time 1
Time_now
Entry time 0
Reentry time 0
Self time 0
Self time 0 + Time_now - Reentry time 0
a0
Return address 1
Table entry index 1
Time_now
Entry time 1
Entry index
Reentry time 1
Time_now
Self time 1
zero
profiler_exit
a4
a0

FIG. 10

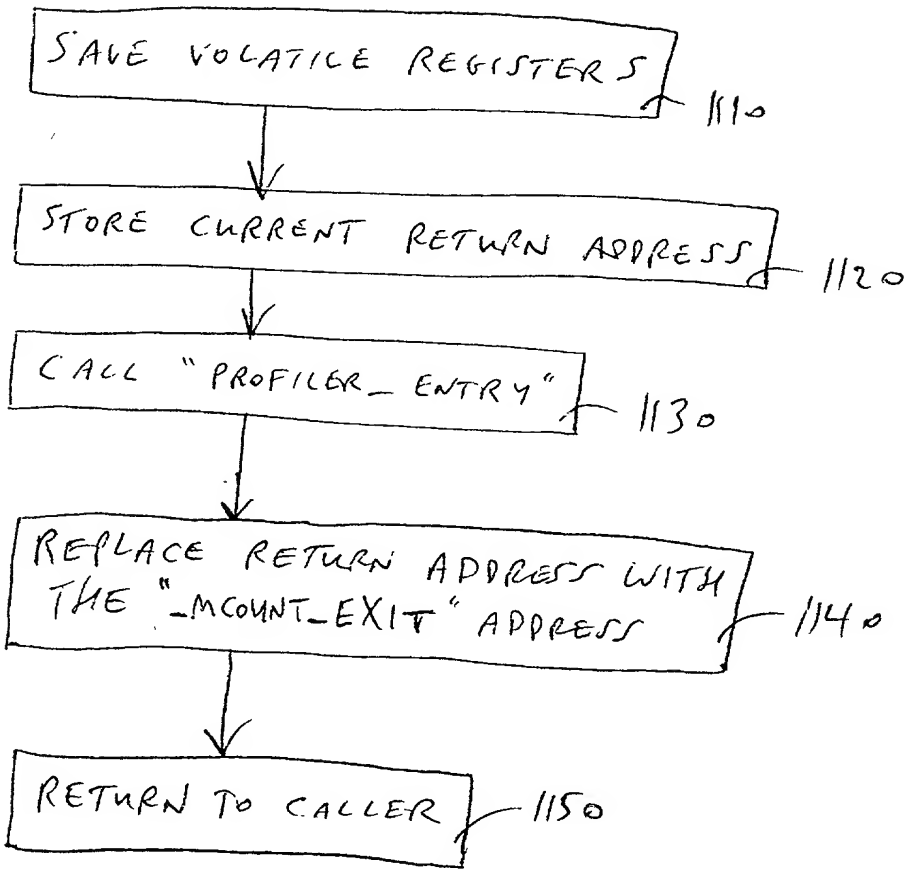

SAVE VOLATILE REGISTERS
1110
STORE CURRENT RETURN ADDRESS
1120
CALL "PROFILER_ENTRY"
1130
REPLACE RETURN ADDRESS WITH THE "_MCOUNT_EXIT" ADDRESS
1140
RETURN TO CALLER
1150

FIG. 11

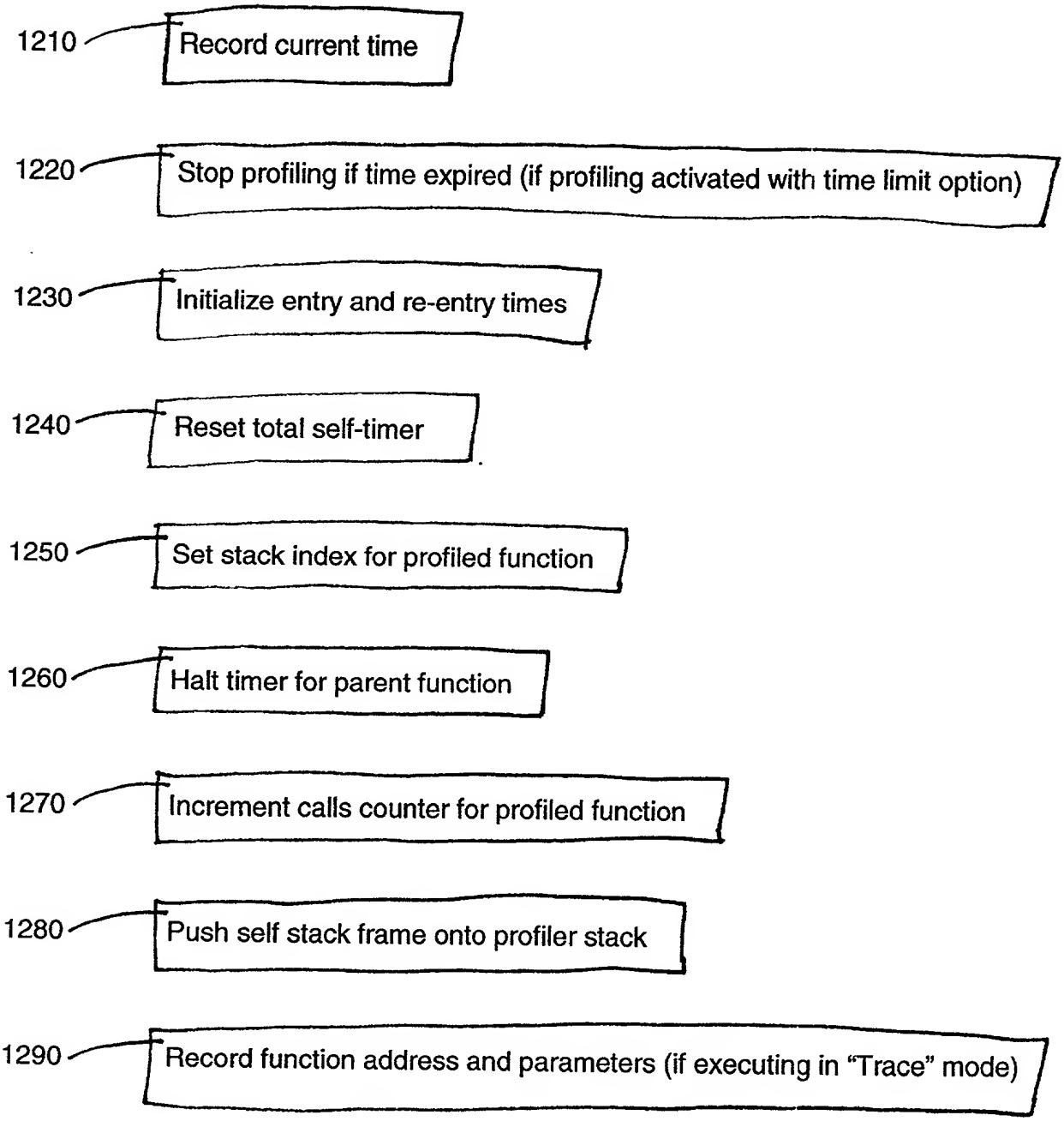

1210
Record current time
1220
Stop profiling if time expired (if profiling activated with time limit option)
1230
Initialize entry and re-entry times
1240
Reset total self-timer
1250
Set stack index for profiled function
1260
Halt timer for parent function
1270
Increment calls counter for profiled function
1280
Push self stack frame onto profiler stack
1290
Record function address and parameters (if executing in "Trace" mode)

FIG. 12

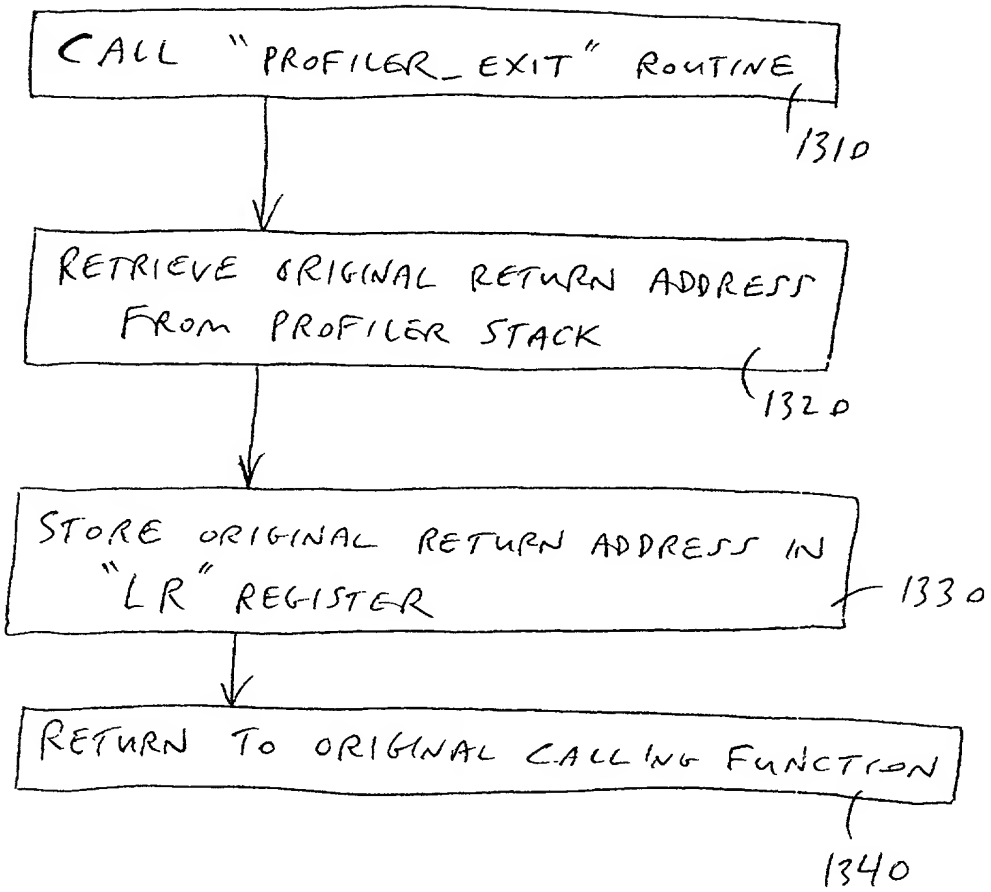

CALL "PROFILER_EXIT" ROUTINE
1310
RETRIEVE ORIGINAL RETURN ADDRESS FROM PROFILER STACK
1320
STORE ORIGINAL RETURN ADDRESS IN "LR" REGISTER
1330
RETURN TO ORIGINAL CALLING FUNCTION
1340

FIG. 13

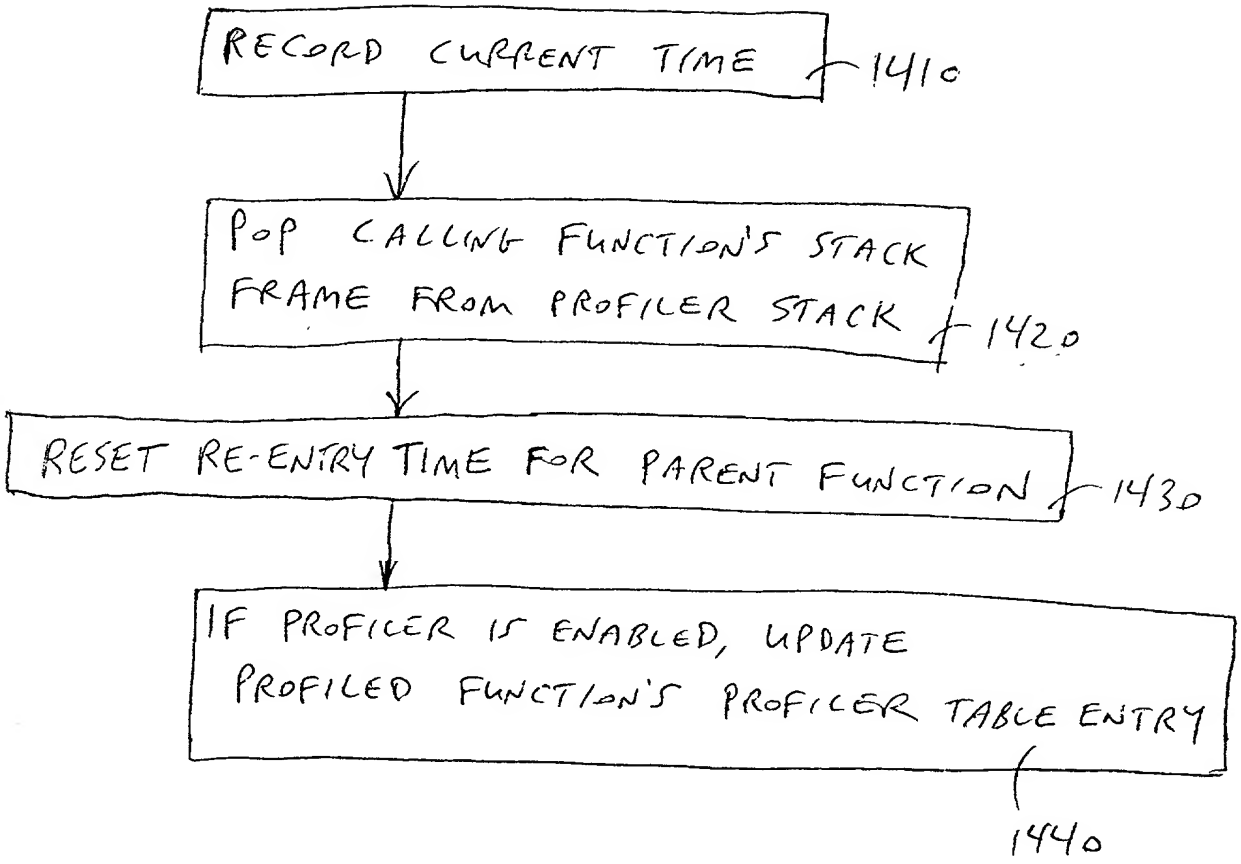

RECORD CURRENT TIME
1410
POP CALLING FUNCTION'S STACK FRAME FROM PROFILER STACK
1420
RESET RE-ENTRY TIME FOR PARENT FUNCTION
1430
IF PROFILER IS ENABLED, UPDATE PROFILED FUNCTION'S PROFILER TABLE ENTRY
1440

FIG. 14

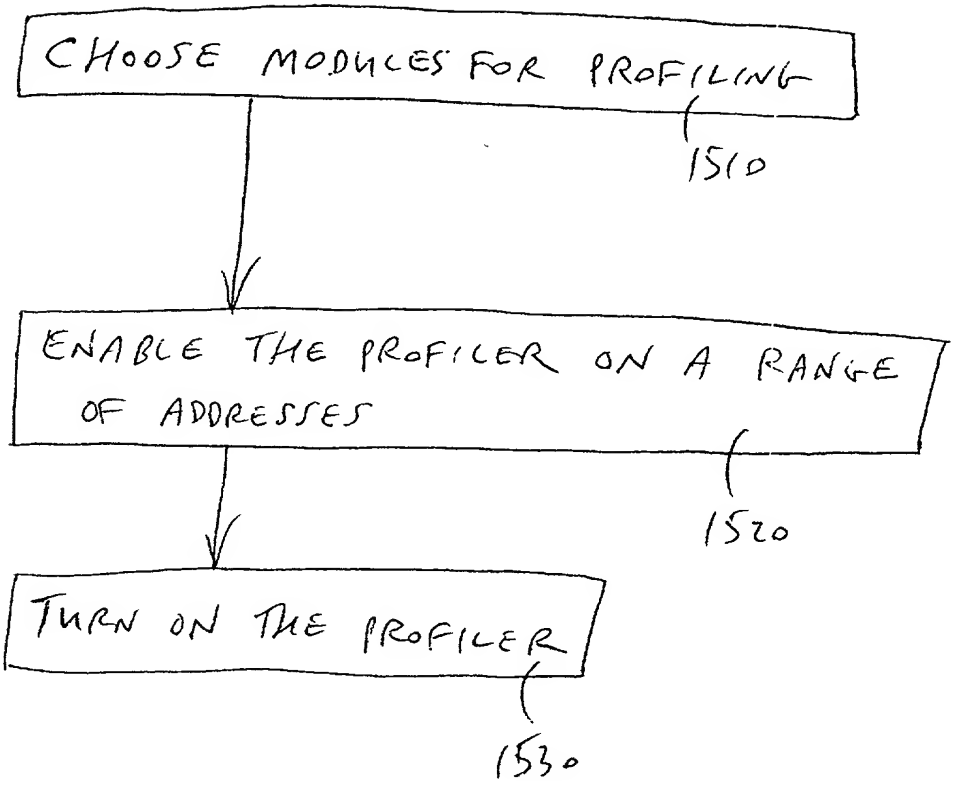

CHOOSE MODULES FOR PROFILING
1510
ENABLE THE PROFILER ON A RANGE OF ADDRESSES
1520
TURN ON THE PROFILER
1530

FIG. 15

```
UTILITY 83 -- Sym-Profiler : TIME: SEP/30/01 18:13:37
-----------------------------

83, [HELP | AAAA], [STUP | EXEC | DISP] - Display help page
                    STUP - Setup
                    EXEC - Runtime control
                    DISP - Display statistics

P in prompt - Profiler in ON
Event trace message: 0xd3
```

FIG. 16A

```
UTILITY 83 -- Sym-Profiler : TIME: SEP/30/01 18:16:4
----------------------------


----------   Profiler Setup   ----------
83,DEF,<range number>,<start-addr>,<end-addr>,<flags>
   Add to <flags>: 0x1 - symbolic addr range
                   0x2 - multi-module range, p6 and p7 = name of end module
83,[C | D | E],<ranges bitmap> - Clear, Disable, Enable ranges per bitmap
83,EE                         - Restore setup from NVD
83,DD,<start range>,<flags>   - Display profiler setup
   Add to <flags>: 0x1 - display from NVD
83,DRAW,<ranges bitmap>,<flags> - Display profiler setup in raw mode
   Add to <flags>: 0x1 - display from NVD
```

FIG. 16 B

```
UTILITY 83 -- Sym-Profiler : TIME: SEP/30/01 18:19:0
----------------------------

----------   Profiler Execution   ----------
83,CA                    - Clear profiler statistics
83,1,<mode>,[<time>] - Start profiling (optionally stop after <time> seconds)
     <mode>  TRAC - Tracing
             PROF - Profiling
             ONCE - Do once then disable (per function)
83,0                     - Stop profiling
```

FIG. 16C

```
UTILITY 83 -- Sym-Profiler : TIME: SEP/30/01 18:20:1
----------------------------

----------   Profiler Display   ----------
83,A,<skip>,<lines>,<key>,<order> - Display statistics
  <skip>  - Number of entries to skip before starting display
  <lines> - Number of lines to display in page (default is 16)

  <key>='CALL' - Rank statistics by number of calls
        'SELF' - Rank statistics by function time alone (no descendant time)
        'TOTL' - Rank statistics by total function time
        'SEAV' - Rank statistics by function time (average)
        'TOAV' - Rank statistics by total function time (average)
  <order>= [ASCE | DESC] - Rank statistics in ascending/descending order
83,AF,0,<function address> - Display statistics for one function
83,AF,1,<module-name>,<module-name>,<offset>  specify function as
module+offset
83,AF,1,<module-name>,<module-name>,1 - Display statistics for all functions
in module
83,AA,<ranges bitmap> - Display all functions in range
83,A0,<ranges bitmap> - Display functions in range which were not called
83,A1,<ranges bitmap> - Display functions in range which were     called
```

FIG. 16D

DA- 16a : S5 IMPL MONITOR > 83,1,50524F4

UTILITY 83 -- Sym-Profiler : TIME: OCT/01/01 11:06:59

Profiler is ON, Mode: Profiling, Setup time: OCT/01/01 11:04:02

No.	Range Start	End	Functions Found	First	Last	Statistics table First	Last
*01.	unzip_ld+00000 100000	sfs_vnop+133e8 500000	1035	00260a5c	004c5088	008d0000	008d8140

Runtime information:
Start time: OCT/01/01 11:06:59; Stop time: ----------
Runtime limit: No limit

FIG. 17

```
DA- 16a : PS5 IMPL MONITOR > 83,0

UTILITY 83 -- Sym-Profiler : TIME: OCT/01/01 11:01:11
----------------------------
Profiler:  OFF

 Start time: 10:56:01 OCT/01/01 0a587783
 Stop  Time: 11:01:11 OCT/01/01 0a5879ee
 Run Time  : 0d 00h 05m 09s
```

FIG. 18

DA- 16a : PS5 IMPL MONITOR > 83,A

UTILITY 83 -- Sym-Profiler : TIME: OCT/01/01 11:14:37

Profiler: ON, MODE: Profiling

Start time: 11:06:59 OCT/01/01 0a587ca6
Run Time : 0d 00h 07m 38s

Routine Module/Offset	Address	Number of calls	Time (micros) Self	Time (micros) Total	Average time (micros) Self	Average time (micros) Total	% time Self	% time Tot.
thermal_+00170	49e22c	1	87911	87911	87911	87911	0	0
thermal_+001a0	49e25c	4	57332	57332	14333	14333	0	0
thermal_+00004	49e0c0	3	483	483	161	161	0	0

Min stack pointer = 0082fba8 at 0049e0c0, stack available = 0002fba8 bytes

FIG. 19A

UTILITY 83 -- Sym-Profiler : TIME: OCT/01/01 14:54:30

Profiler: ON, MODE: Profiling

Start time: 14:54:23 OCT/01/01 0a58e73e
Run Time : 0d 00h 00m 07s

Routine Module/Offset	Address	Number of calls	Time (micros) Self	Time (micros) Total	Average time (micros) Self	Average time (micros) Total	% time Self	% time Tot.
daschedu+01cc0	3641d0	44501	226182	5145138	5	115	3	73
daschedu+00ad8	362fe8	32752	98552	3948230	3	120	1	56
dawrites+00004	267ec0	32752	1593660	3849616	48	117	22	54
dawrites+016c0	26957c	127683	1400634	2053441	10	16	20	29
dawrites+012d0	26918c	488521	515448	605265	1	1	7	8
daschedu+01048	363558	46165	402229	467912	8	10	5	6
daschedu+01598	363aa8	44501	205113	455331	4	10	2	6
daschedu+007dc	362cec	11804	19880	447482	1	37	0	6
daprefch+03788	2641d0	11804	23263	429399	1	36	0	6
daprefch+01d0c	262754	11804	74382	376237	6	31	1	5
daprefch+01fa8	2629f0	46916	283656	309941	6	6	4	4
daschedu+00004	362514	44501	173907	205722	3	4	2	2
da_utils+02858	2f1468	44339	163917	163917	3	3	2	2
dawrites+00dd4	268c90	133836	133836	133836	1	1	1	1
dawrites+00728	2685e4	41789	124541	125498	2	3	1	1
daschedu+00e04	363314	2521	109413	109413	43	43	1	1

Min stack pointer = 0082f260 at 0026c4f4, stack available = 0002f260 bytes

DA- 2a : P@sS5 DISK SUBSYSTEM:>

Set-Up-Time: OCT/01/01 14:56:17 Stop-Trigger: FFFFFFFF Phase: COLLECT
Collect-Time: OCT/01/01 14:56:17 GM Reject: 0(0)/0
Util 97 Info - Setup Time: OCT/01/01 13:22:25

DONE.
DA- 2a : P.sS5 DISK SUBSYSTEM:>

FIG. 19B

Util_end: 00000000 msgid=552905
10/01/01 15:22:06 15b -> PC (Unknown)

Util_end: 00000000 msgid=CEFABEBA
10/01/01 15:22:06 01a -> PC (Unknown)

Util_end: 00000000 msgid=CEFABEBA
10/01/01 15:22:06 15a -> PC (Unknown)

Util_end: 00000000 msgid=CEFABEBA
10/01/01 15:22:06 01b -> PC (Unknown)

Util_end: 00000000 msgid=CEFABEBA
10/01/01 15:22:06 16a -> PC (Unknown)

Util_end: 00000000 msgid=CEFABEBA
10/01/01 15:22:06 16d -> PC (Unknown)

Util_end: 00000000 msgid=CEFABEBA
10/01/01 15:22:06 16b -> PC (Unknown)

Util_end: 00000000 msgid=CEFABEBA
10/01/01 15:22:06 16c -> PC (Unknown)

Util_end: 00000000 msgid=CEFABEBA
10/01/01 15:23:03 02b -> PC (Unknown)

OCT/01/01 14:57:17 CONTROL STORE TEST...01 add(00080000) .....PASSED.

DA- 2b : .sS5 DISK SUBSYSTEM:>
10/01/01 15:23:20 01b -> PC (Unknown)

OCT/01/01 14:57:34 CONTROL STORE TEST...01 add(00080000) .....PASSED.

DA- 1b : .sS5 DISK SUBSYSTEM:>
10/01/01 15:24:03 01a -> PC (Unknown)

FIG. 19C

DA- 1a : .sS5 DISK SUBSYSTEM:>
10/01/01 15:26:48 02b -> PC (Unknown)
ptErrorResp [85.5F16.06] at 10/01/2001 3:26:48 PM
10/01/01 15:26:48 01a -> PC (Unknown)

DA- 1a : .sS5 DISK SUBSYSTEM:>
10/01/01 15:30:09 Manual Inlines command sent: 83,A
10/01/01 15:30:09 PC (INLINES) -> 02a,,,

Command to dir 02a: [83,A,] msgid=552A05
10/01/01 15:30:09 02a -> PC (INLINES)

UTILITY 83 -- Sym-Profiler : TIME: OCT/01/01 15:04:23

Profiler: ON, MODE: Profiling

Start time: 14:54:23 OCT/01/01 0a58e73e
Run Time : 0d 00h 10m 00s

Routine Module/Offset	Address	Number of calls	Time (micros) Self	Total	Average time (micros) Self	Total	% time Self	Tot.
daschedu+01cc0	3641d0	2522952	12991240	329169241	5	130	2	54
daschedu+00ad8	362fe8	2055489	6183238	262200235	3	127	1	43
dawrites+00004	267ec0	2055489	109371204	256003627	53	124	18	42
dawrites+016c0	26957c	6998173	85742065	126349753	12	18	14	21
dawrites+012d0	26918c	31979335	30528363	34969248	0	1	5	5
daschedu+01048	363558	2685627	24248920	28323741	9	10	4	4
daschedu+01598	363aa8	2522952	11494217	26585146	4	10	1	4
da_copyt+02970	37b428	2652	35059	14419079	13	5437	0	2
daschedu+007dc	362cec	387932	702492	13111048	1	33	0	2
daschedu+00004	362514	2522952	10933652	12611157	4	4	1	2
daprefch+03788	2641d0	387932	820965	12478041	2	32	0	2
daprefch+01d0c	262754	387932	2333514	11458184	6	29	0	1
daschedu+00828	362d38	54643	363880	10799259	6	197	0	1
daschedu+00a10	362f20	7919	812005	10421421	102	1316	0	1

FIG. 19D

da_rtask+00004 373224 | 3503| 50487| 9518913| 14| 2717| 0| 1|
dawrites+00728 2685e4 | 2723405| 9344426| 9422713| 3| 3| 1| 1|
Min stack pointer = 00825370 at 002a6450, stack available = 00025370 bytes

DA- 2a : P.sS5 DISK SUBSYSTEM:>
Util_end: 00000000 msgid=552A05
10/01/01 15:31:47 15a -> PC (Unknown)

OCT/01/01 15:06:01 CONTROL STORE TEST...01 add(00080000) .....PASSED.
10/01/01 15:32:03 16a -> PC (Unknown)

OCT/01/01 15:06:17 CONTROL STORE TEST...01 add(00080000) .....PASSED.
10/01/01 15:33:55 15b -> PC (Unknown)

OCT/01/01 15:08:09 CONTROL STORE TEST...01 add(00080000) .....PASSED.
10/01/01 15:34:01 02a -> PC (INLINES)
ptErrorResp [85.5F16.06] at 10/01/2001 3:34:01 PM
10/01/01 15:34:01 01a -> PC (Unknown)

DA- 1a : .sS5 DISK SUBSYSTEM:>
10/01/01 15:34:17 16b -> PC (Unknown)

OCT/01/01 15:08:31 CONTROL STORE TEST...01 add(00080000) .....PASSED.
10/01/01 15:34:58 Closing Inlines
10/01/01 15:35:03 Opening Inlines, user: , group: PC group
10/01/01 15:35:17 Manual Inlines command sent: 83,AAAA
10/01/01 15:35:17 PC (INLINES) -> 01a,,,

Command to dir 01a: [83,AAAA,] msgid=552C05
10/01/01 15:35:17 01a -> PC (INLINES)

UTILITY 83 -- Sym-Profiler : TIME: OCT/01/01 15:09:31

83, [HELP | AAAA], [STUP | EXEC | DISP] - Display help page
STUP - Setup

FIG. 19E

EXEC - Runtime control
DISP - Display statistics

P in prompt - Profiler in ON
Event trace message: 0xd3

DA- 1a : .sS5 DISK SUBSYSTEM:>
Util_end: 00000000 msgid=552C05
10/01/01 15:35:22 Manual Inlines command sent: .
10/01/01 15:35:22 PC (INLINES) -> 01b,,

Command to dir 01b: . msgid=552D05
10/01/01 15:35:22 01b -> PC (INLINES)

.
NON HEX CHAR

DA- 1b : .sS5 DISK SUBSYSTEM:>
10/01/01 15:35:26 Manual Inlines command sent: .
10/01/01 15:35:26 PC (INLINES) -> 02a,,,

Command to dir 02a: . msgid=552E05
10/01/01 15:35:26 02a -> PC (INLINES)

.
NON HEX CHAR

DA- 2a : P.sS5 DISK SUBSYSTEM:>
10/01/01 15:35:31 Manual Inlines command sent: 83,A
10/01/01 15:35:31 PC (INLINES) -> 02a,,,

```
Command to dir 02a: [83,A,] msgid=552F05
10/01/01 15:35:32  02a -> PC (INLINES)


UTILITY 83 -- Sym-Profiler : TIME: OCT/01/01 15:09:45
---------------------------
Profiler:  ON, MODE: Profiling
```

FIG. 19F

Start time: 14:54:23 OCT/01/01 0a58e73e
Run Time : 0d 00h 15m 22s

Routine Module/Offset	Address	Number of calls	Time (micros) Self	Time (micros) Total	Average time (micros) Self	Average time (micros) Total	% time Self	% time Tot.
daschedu+01cc0	3641d0	4048572	21154575	494195505	5	122	2	53
daschedu+00ad8	362fe8	3291389	9902877	385479049	3	117	1	41
dawrites+00004	267ec0	3291389	158197598	375574540	48	114	17	40
dawrites+016c0	26957c	9793496	127646078	187230604	13	19	13	20
dawrites+012d0	26918c	47216195	44602356	51039823	0	1	4	5
daschedu+01048	363558	4491138	40201730	46996117	8	10	4	5
daschedu+01598	363aa8	4048572	18782025	42846503	4	10	2	4
da_copyt+02970	37b428	4805	80589	21817204	16	4540	0	2
daschedu+007dc	362cec	746474	1320046	21643679	1	28	0	2
daprefch+03788	2641d0	746474	1538576	20458191	2	27	0	2
daschedu+00004	362514	4048572	17360936	19983269	4	4	1	2
daprefch+01d0c	262754	746474	3842503	18513306	5	24	0	2
daschedu+00828	362d38	94399	677156	17337392	7	183	0	1
da_copyt+020e4	37ab9c	4805	1255412	17013818	261	3540	0	1
daschedu+00a10	362f20	15427	1584969	16634358	102	1078	0	1
da_rtask+00004	373224	11742	133494	16066053	11	1368	0	1

Min stack pointer = 00823048 at 0049f394, stack available = 00023048 bytes

DA- 2a : P.sS5 DISK SUBSYSTEM:>

FIG. 19G

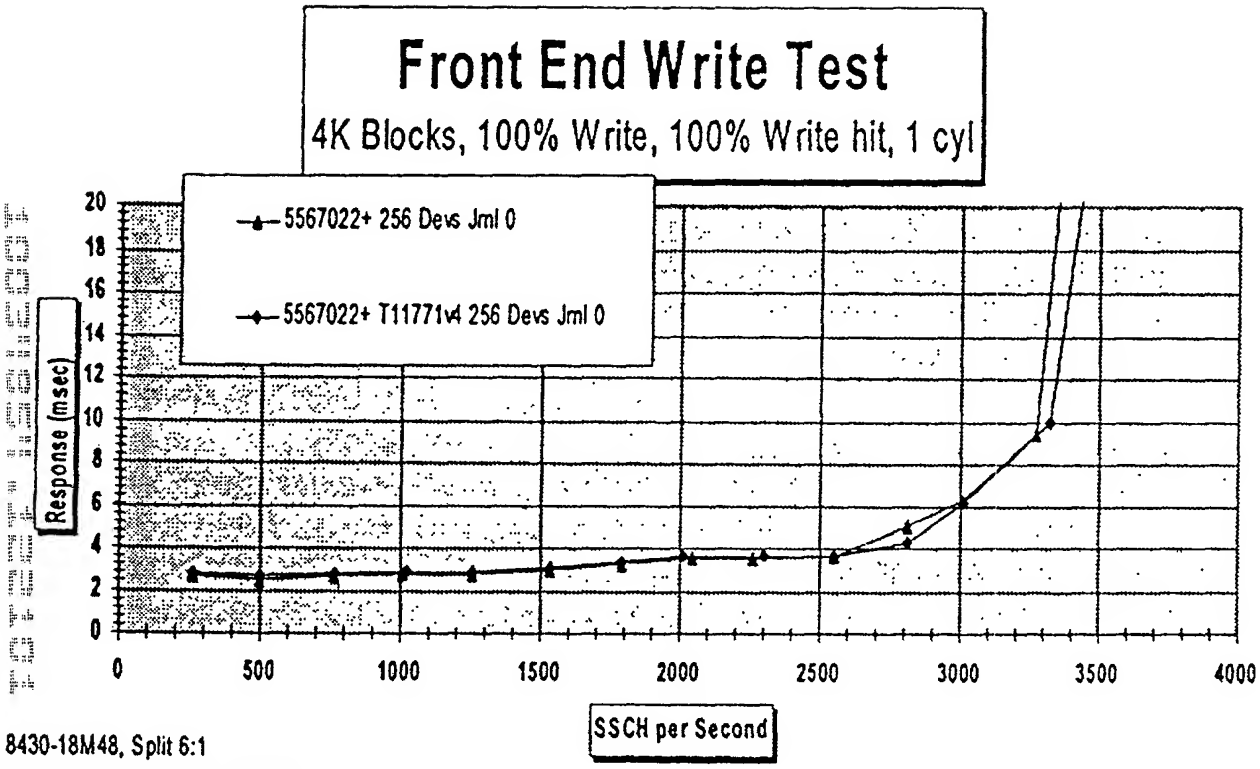
Profiler: ON, MODE: Profiling

Start time: 14:54:23 OCT/01/01 0a58e73e
Run Time : 0d 00h 35m 31s

Routine Module/Offset	Address	Number of calls	Time (micros) Self	Time (micros) Total	Average time (micros) Self	Average time (micros) Total	% time Self	% time Tot.
dawrites+00004	267ec0	8159810	379921296	906208848	46	111	17	42
dawrites+016c0	26957c	23194433	311286293	454524217	13	19	14	21
dawrites+012d0	26918c	113817075	107355424	123512544	0	1	5	5
daschedu+01048	363558	10837699	97770758	114282199	9	10	4	5
daschedu+01cc0	3641d0	9992748	51666137	1170386199	5	117	2	54
daschedu+01598	363aa8	9992748	46125615	103420081	4	10	2	4
daschedu+00004	362514	9992748	41495305	47383227	4	4	1	2
da_utils+02858	2f1468	9958730	37412992	37412992	3	3	1	1
dawrites+00728	2685e4	10299487	33491913	33763773	3	3	1	1
daschedu+00e04	363314	655745	28488210	28488210	43	43	1	1
daprefch+01fa8	2629f0	4101228	24844129	27249016	6	6	1	1
daschedu+00ad8	362fe8	8159810	24544745	930763270	3	114	1	43
da_copyt+00100	378bb8	10118	23769100	25413369	2349	2511	1	1
dawrites+00dd4	268c90	23484807	23485140	23485140	1	1	1	1
idle_stt+00080	394434	10995564	21948672	21948761	1	1	1	1
cache_rt+03f04	26e8e4	2313095	19836710	23451890	8	10	0	1

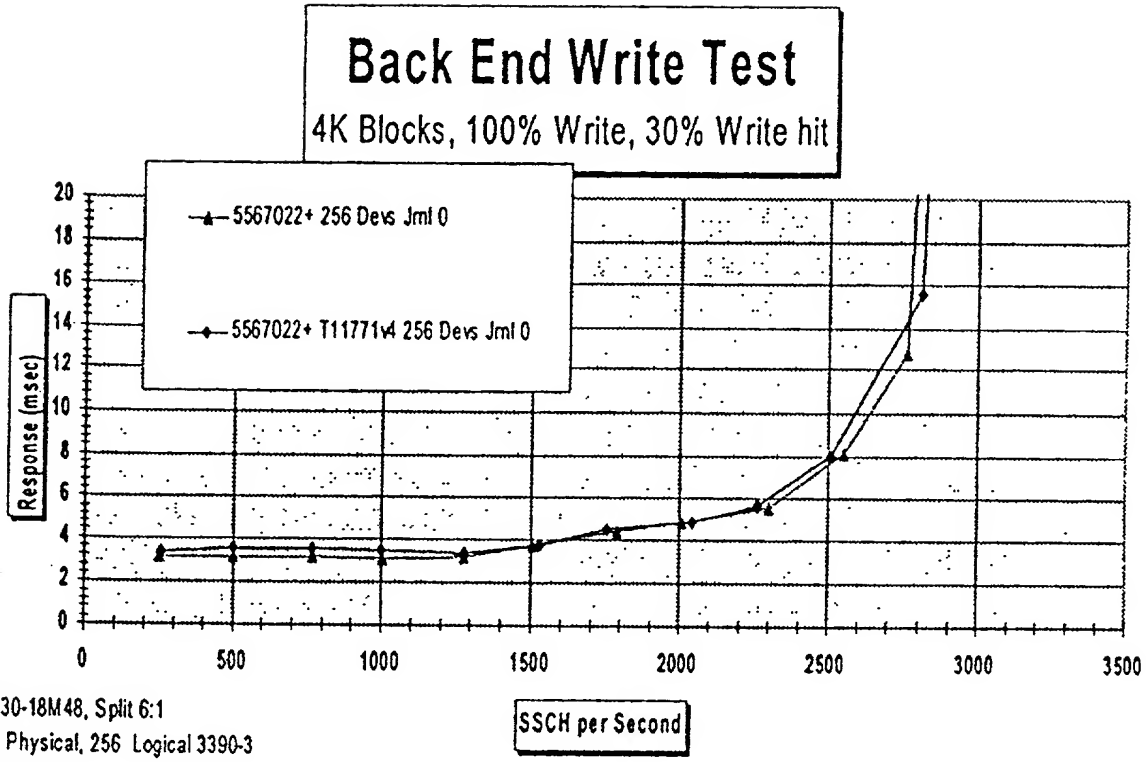
Min stack pointer = 00823048 at 0049f394, stack available = 00023048 bytes

FIG. 20



8430-18M48, Split 6:1
96 Physical, 256 Logical 3390-3
8GB Cache
8 Host Channels

FIG. 21



8430-18M48, Split 6:1
96 Physical, 256 Logical 3390-3
8GB Cache
8 Host Channels

FIG. 22

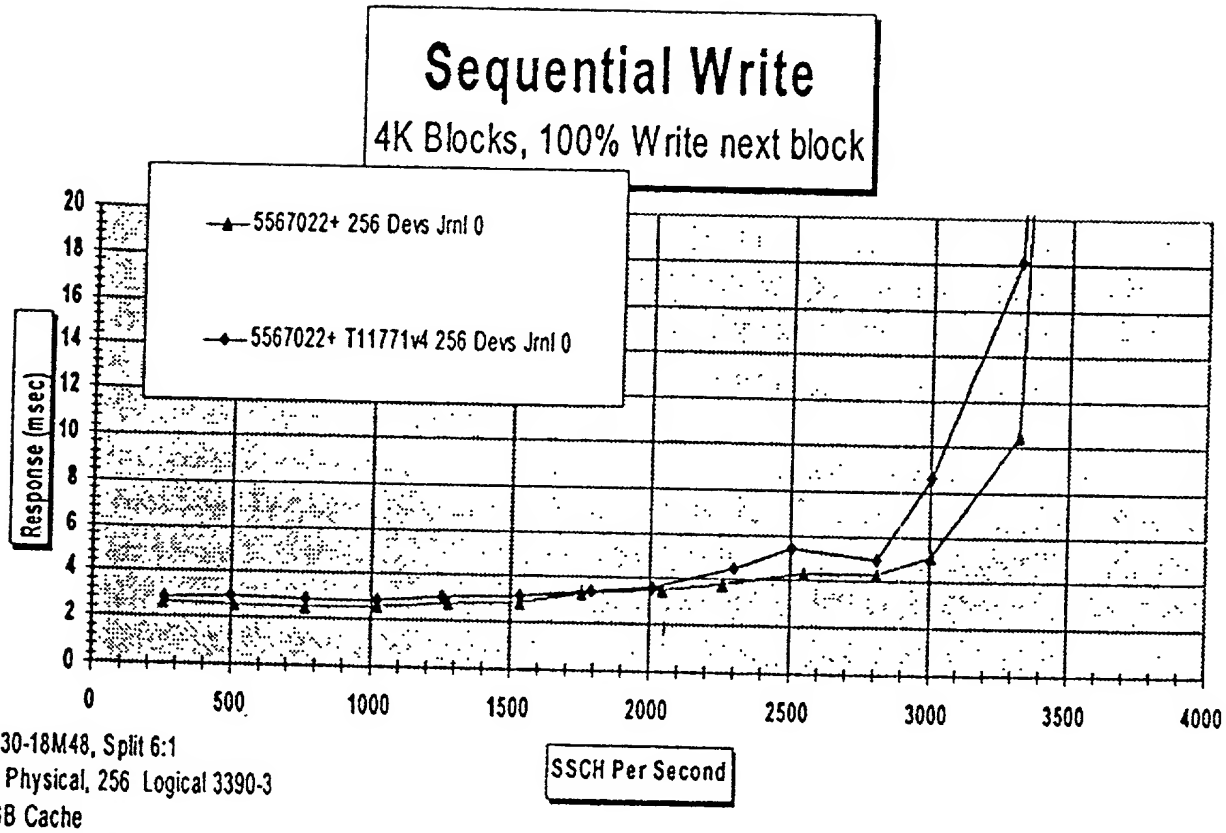

Sequential Write
4K Blocks, 100% Write next block
5567022+ 256 Devs Jrnl 0
5567022+ T11771v4 256 Devs Jrnl 0
Response (msec)
0
2
4
6
8
10
12
14
16
18
20
0
500
1000
1500
2000
2500
3000
3500
4000
SSCH Per Second
8430-18M48, Split 6:1
96 Physical, 256 Logical 3390-3
8GB Cache
8 Host Channels

FIG. 23.

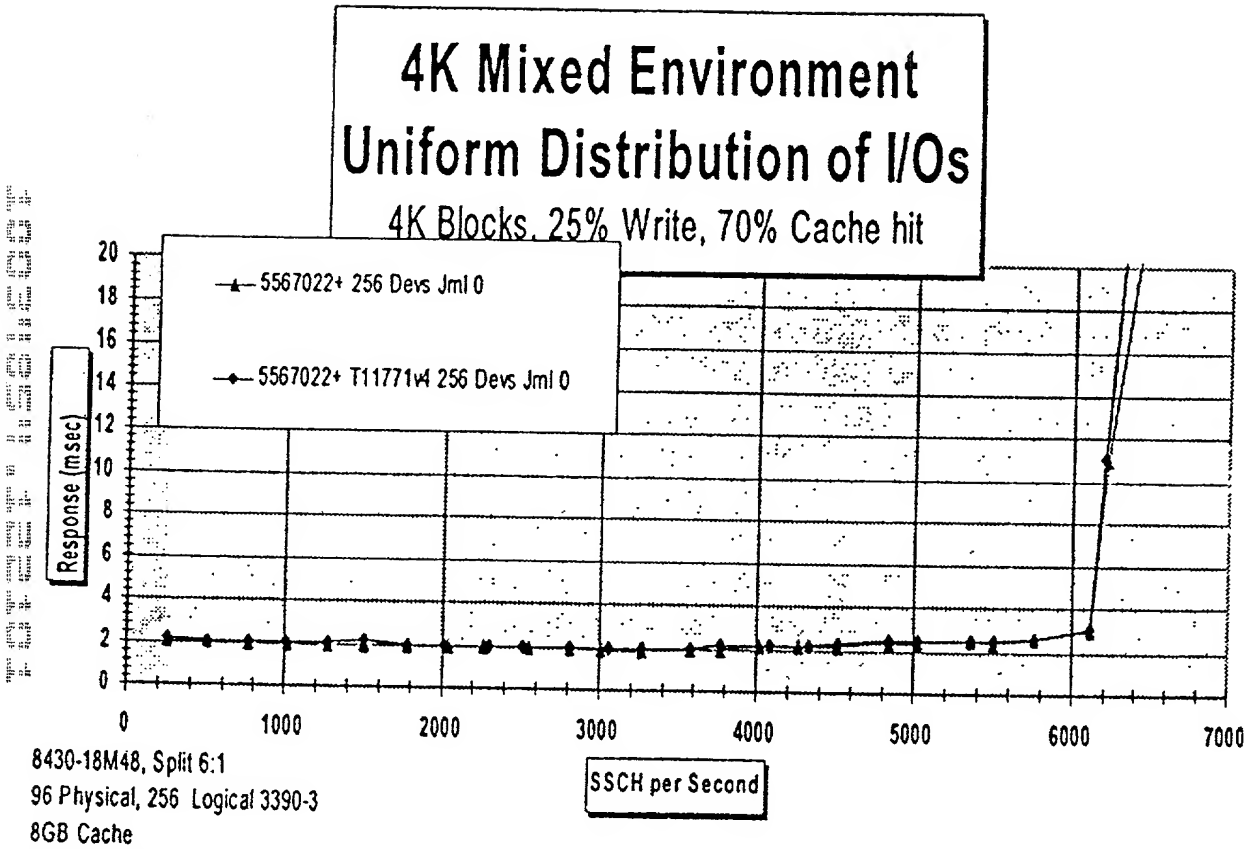
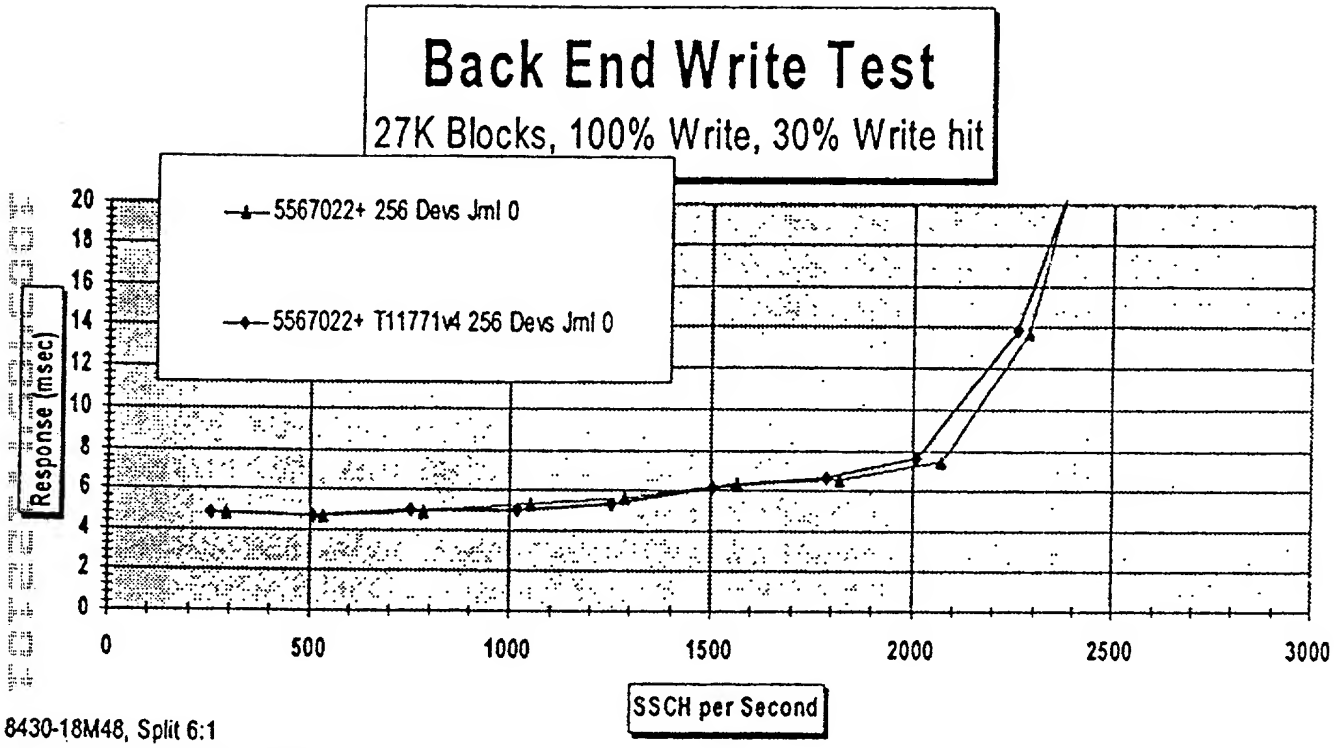
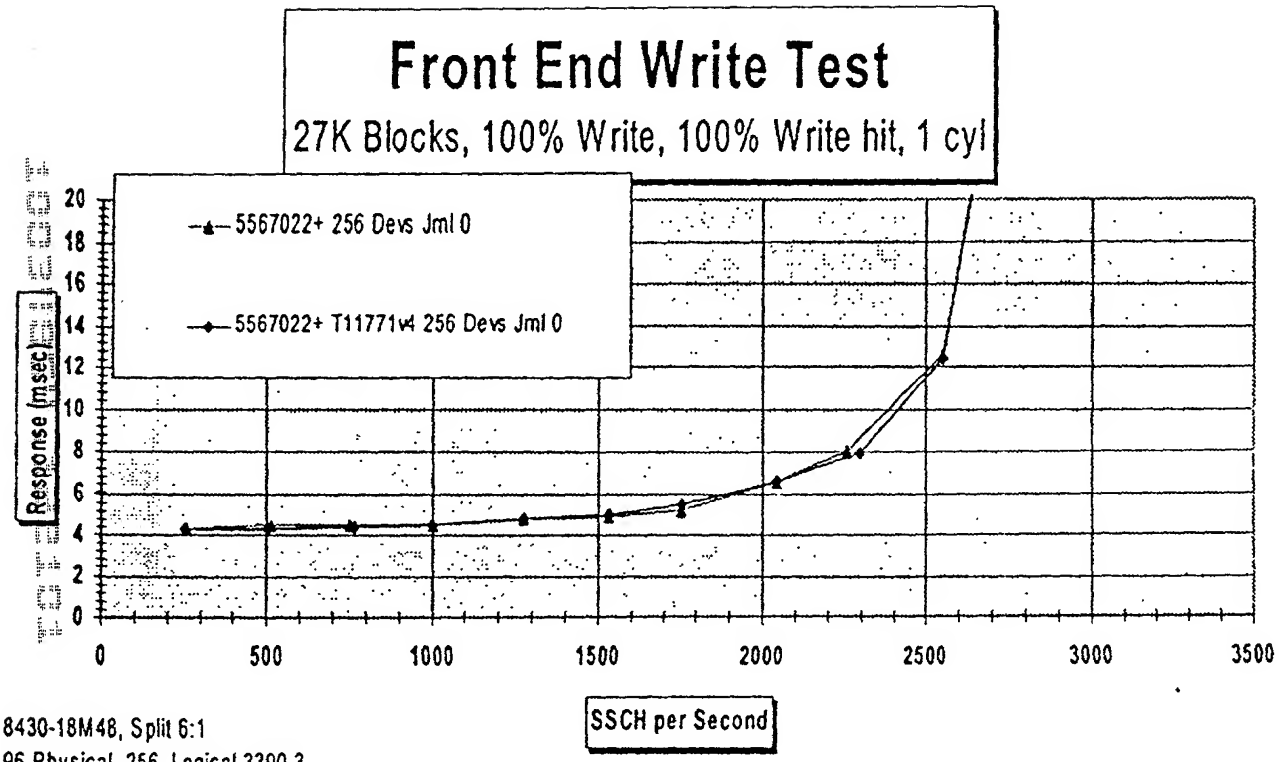

4K Mixed Environment
Uniform Distribution of I/Os
4K Blocks, 25% Write, 70% Cache hit
5567022+ 256 Devs Jml 0
5567022+ T11771v4 256 Devs Jml 0
Response (msec)
0
2
4
6
8
10
12
14
16
18
20
0
1000
2000
3000
4000
5000
6000
7000
SSCH per Second
8430-18M48, Split 6:1
96 Physical, 256 Logical 3390-3
8GB Cache
8 Host Channels

FIG. 24



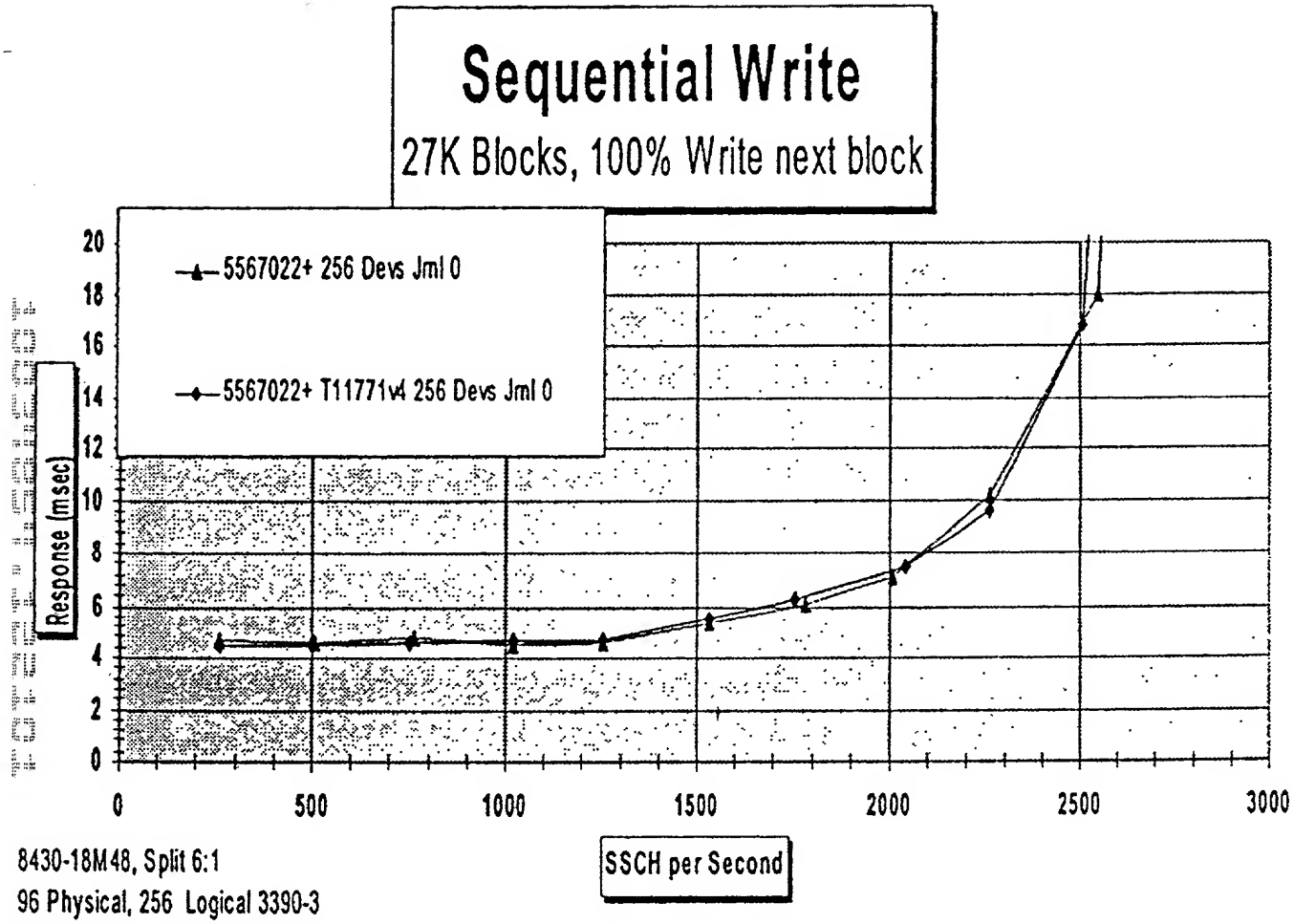
8430-18M48, Split 6:1
96 Physical, 256 Logical 3390-3
8GB Cache
8 Host Channels

FIG. 25



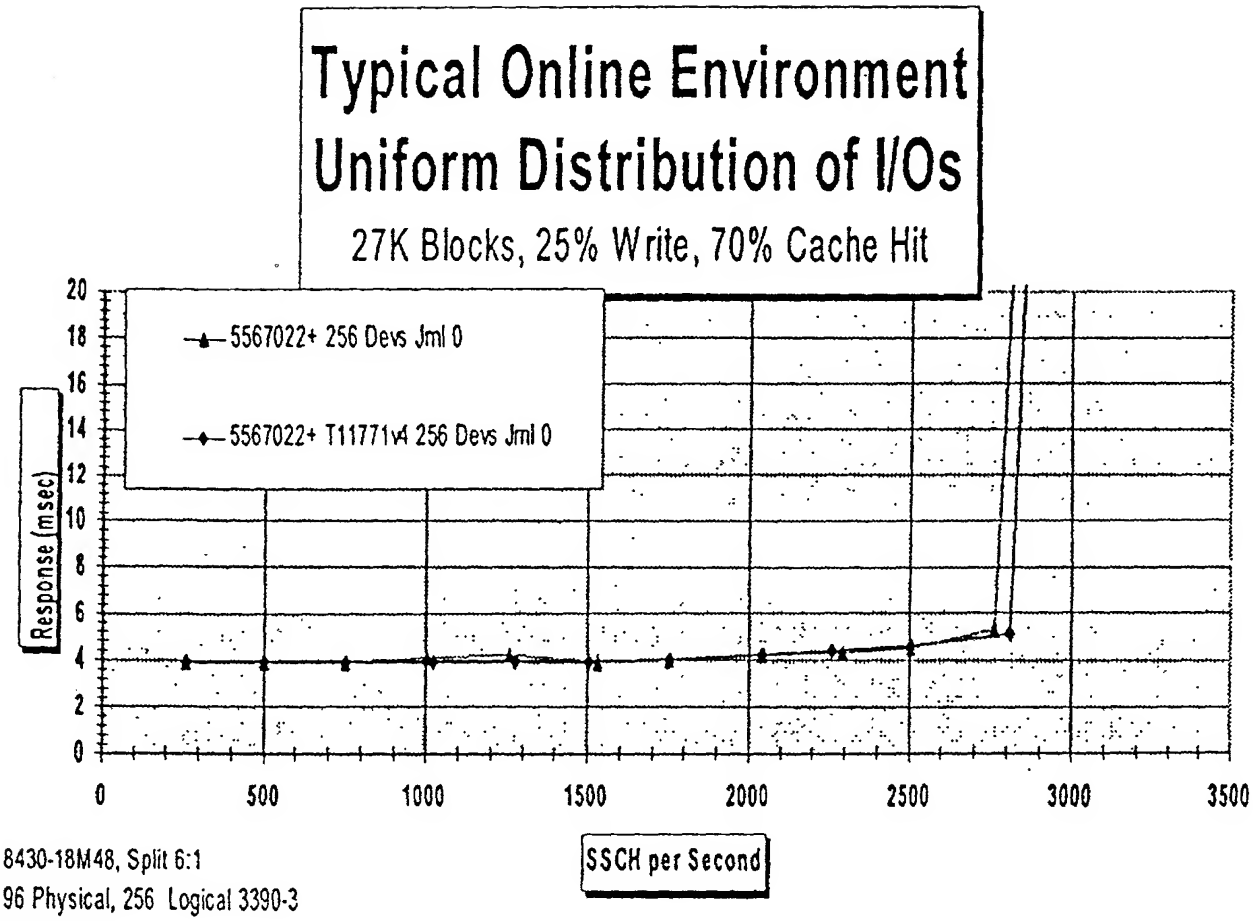
8430-18M48, Split 6:1
96 Physical, 256 Logical 3390-3
8GB Cache
8 Host Channels

FIG. 26



8430-18M48, Split 6:1
96 Physical, 256 Logical 3390-3
8GB Cache
8 Host Channels

FIG. 27



8430-18M48, Split 6:1
96 Physical, 256 Logical 3390-3
8GB Cache
8 Host Channels

FIG. 28